

APPLICANT(S): HAVIV, Yaron ; COREM, Guy
SERIAL NO.: 09/863,423
FILED: May 24, 2001
Page 2

REMARKS

The present response is intended to be fully responsive to all points of rejection raised by the Examiner and is believed to place the application in condition for allowance. Favorable reconsideration and allowance of the application is respectfully requested.

Applicants assert that the present invention is new, non-obvious and useful. Prompt consideration and allowance of the claims is respectfully requested.

Status of Claims

Claims 1 through 20 are pending in the application. Claims 1 through 20 have been rejected.

Claim Rejections

35 U.S.C. § 102 Rejections

In the Office Action, the Examiner rejected Claims 1-5, 8-12, 15, and 18 under 35 U.S.C. § 102(e) as being anticipated by Ramberg, et al. (US 6,398,105). Applicants respectfully traverse this rejection in view of the remarks that follow.

The '105 patent neither teaches nor suggest all the limitations of independent claims 1, 8, 15 and 18. Independent claims 1, 8, 15 and 18 each include the limitations of:

1. a communication interface adapted to bypasses a substantial portion of the operating system kernel; and
2. providing filtered application data directly to communication hardware by bypassing a substantial portion of the operating system kernel.

APPLICANT(S): HAVIV, Yaron ; COREM, Guy
SERIAL NO.: 09/863,423
FILED: May 24, 2001
Page 3

Completely unrelated to the claimed limitations, the '105 reference generally teaches:

"A method and system for intelligently routing data received from an automatic data collection ("ADC") device in an ADC device platform based on its type. A data routing mechanism operates on the data-receiving side of an ADC data server. After identifying the characteristics of the input data, the data routing mechanism determines the destination for the data based on the characteristics, then routes the data to the selected destination. For some types of data, the selected destination may be an intermediate destination where the data undergoes additional processing before being transmitted to another location, while for other types of data the selected destination may be the application that ultimately processes the data. For example, the data routing mechanism may receive a set of input data, analyze the data to determine that the data is voice data, and then route the data to a speech recognition module that processes voice data. ADC devices accommodated by the system include bar code readers, speech recognition systems, RF tag readers, resonator readers, and two-dimensional symbol readers optical character recognition ("OCR") systems. The invention finds application within a network of ADC device platforms that receive requests for input data from both local and remote applications. Data may be communicated to remote users using any data protocol, including the Transmission Control Protocol ("TCP"), the User Datagram/Internet Protocol ("UDP/IP") or the User Datagram Plus Protocol ("UDP+")."

The Examiner supports her position that the above two limitation are disclosed in the '105 patent by stating:

"Ramberg, et al. discloses a method for filtered application-to-application communication of applications running on computing platforms including an operating system kernel, said method comprising: providing a communication interface to an application [COL.6, lines 23-31], wherein the communication

APPLICANT(S): HAVIV, Yaron ; COREM, Guy
SERIAL NO.: 09/863,423
FILED: May 24, 2001
Page 4

interface bypasses a substantial portion of the operating system kernel; [COL.5, lines 49-53] filtering application data received from a process of said application according to a predetermined policy; and [COL.5, lines 22-23 and col.8, lines 50-55] providing said filtered application data directly to communication hardware [COL.16, lines 32-35] by bypassing a substantial portion of the operating system kernel. [COL.7, lines 54-60 and col.10, lines 20-30]. **The claimed bypassing a substantial portion of the operating system kernel is relative because substantial portion fails to indicate the exact type or amount or quantity that is to bypass (col.6, lines 49-53). Thus, Ramberg teaches broadly claimed bypassing a substantial portion of the operating system where the computing system uses a non-Windows operating system and uses a TCP/IP sockets interface. Ramberg discloses the communication interface such as Winsock socket's interface over TCPIP which Winsock is an application programming interface (API) that provides TCP/IP socket interface (col.6, lines 2331). The applications retrieves and sends data to other applications (col.6, lines 33-35 and col.7, lines 55-60) where the communication uses interfaces to send information (col.5, lines 50-51), receives data, and provide functionality for adjusting specific device attributes (col.10, lines 20-46). Therefore, the data is provided directly to communication hardware and is without the operating system kernel intervention."**

With all due respect to the Examiner's understanding of the cited '105 references, Applicants wish to point out that although the '105 reference does disclose the possible use of "a non-Windows operating system," and that "communication interface such as Winsock socket's interface over TCP/IP which Winsock is an application programming interface (API) that provides TCP/IP socket interface (col.6, lines 2331)," these facts are not relevant with respect to the claimed limitations. First of all, the fact that the computing platform taught in the '105 reference may use a non-windows operating system, does not

APPLICANT(S): HAVIV, Yaron ; COREM, Guy
SERIAL NO.: 09/863,423
FILED: May 24, 2001
Page 5

mean it doesn't use any operating system. Actually, the '105 reference teaches that "The computing system 120 in the ADC device platform 100 may utilize any suitable operating system. In a preferred embodiment, the computing system utilizes a WindowsCE operating system. The computing system 120 includes local applications 111-113 and an ADC data server 130. The computing system 120 may include more than three local applications, as indicated by the ellipsis between the local application 111 and the local application 112. " Thus, the proposition that if a computing system is using a non-windows operating system, it uses no operating system at all is more than a bit unrealistic. Furthermore, nowhere in the '105 reference does it suggest that the computing platform bypasses an portion of the operating system when communicating with communication hardware.

Although the '105 reference does teach a "communication interface such as Winsock socket's interface over TCP/IIP which Winsock is an application programming Interface (API) that provides TCP/IIP socket interface (col.6, lines 2331)." Applicants take this opportunity to bring the Examiner's attention to the definition of a Winsock sockets (definition is from a Winsock specification found at www.sockets.com/winsock.htm):

"What is Windows Sockets: The Windows Sockets specification defines a network programming interface for Microsoft Windows which is based on the "socket" paradigm popularized in the Berkeley Software Distribution (BSD) from the University of California at Berkeley. It encompasses both familiar Berkeley socket style routines and a set of Windows-specific extensions designed to allow the programmer to take advantage of the message-driven nature of Windows.

Berkeley Sockets: The Windows Sockets Specification has been built upon the Berkeley Sockets programming model which is the de facto standard for TCP/IP networking. It is intended to provide a high degree of familiarity for programmers who are used to programming with sockets in UNIX and other environments, and to simplify the task of porting existing sockets-based source code. The Windows Sockets API is consistent with release 4.3 of the Berkeley Software Distribution (4.3BSD).

APPLICANT(S): HAVIV, Yaron ; COREM, Guy
SERIAL NO.: 09/863,423
FILED: May 24, 2001
Page 6

Basic concepts: The basic building block for communication is the socket. A socket is an **endpoint of communication to which a name may be bound**. Each socket in use has a type and an associated process. Sockets exist within communication domains. A communication domain is an abstraction introduced to bundle common properties of threads communicating through sockets. Sockets normally exchange data only with sockets in the same domain (it may be possible to cross domain boundaries, but only if some translation process is performed). The Windows Sockets facilities support a single communication domain: the Internet domain, which is used by processes which communicate using the Internet Protocol Suite. (Future versions of this specification may include additional domains.)"

As should be clear from the definition of a Winsock socket interface, it is no different from any other Application Program Interface ("API") which may provide an application access to a computing system's communication resource **through the operating system within which the given application is running**. Since the '105 reference teaches that "In a preferred embodiment, remote ADC clients communicate with the ADC data server 130 using the Winsock 1.1 socket's interface over TCP/IP. Winsock is an application programming interface ("API") that provides a TCP/IP socket interface in the Windows operating system. Embodiments of the network communications unit 221 may utilize a variety of communications methods in communicating with remote applications, including sockets, TCP/IP, UDP, and UDP+," it would appear that regardless of the extent the operating system is bypassed, the '105 reference teaches away from the limitations claimed in independent claims 1, 8, 15 and 18, namely bypassing a portion of the operating system.

Therefore, Applicants respectfully request reconsideration and withdrawal of the rejections of independent claims 1, 8, 15 and 18, and all the claims which depend on them.

APPLICANT(S): HAVIV, Yaron ; COREM, Guy
SERIAL NO.: 09/863,423
FILED: May 24, 2001
Page 7

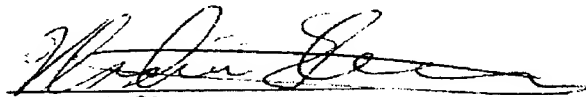
35 U.S.C. § 103 Rejections

In the Office Action, the Examiner rejected dependent claims 6-7, 13-14, 16-17, and 19-20 under 35 U.S.C.103(a), as being unpatentable over Ramberg, et al. (US 6,398,105), and further in view of Bunton, et al. (US 6,690,757).

Applicants respectfully traverse the rejection because a prima facie case of obviousness has not been established. For the same reasons as stated above, namely the Examiner's misunderstanding of the '105 reference, Applicants respectfully traverse the rejection of these claims. More specifically, the limitation of bypassing a substantial portion of the operating system (whether it be windows based or non-windows based) is neither taught nor suggested by the use of a Winsock socket. Furthermore, since all the claims rejected under section 103 are dependent claims, Applicants respectfully assert that these claims are allowable by virtue of their dependence on allowable base claims 1, 8, 15 and 18.

In view of the foregoing remarks, the pending claims are considered to be allowable. Their favorable reconsideration and allowance is respectfully requested.

Respectfully submitted,



Vladimir Sherman
Attorney for Applicant(s)
Registration No. 43,116

Dated: August 25, 2005

Eitan Law Group
C/O Landon-IP
1700 Diagonal Road, Suite 450
Alexandria, VA 22314